

2

# NAVAL POSTGRADUATE SCHOOL

## Monterey, California

AD-A247 321



# THESIS

DTIC  
ELECTE  
MAR 12 1992  
S B D

ELECTRONIC FILING EVALUATION  
OF TAX PROGRAMS

by

Gretchen Ochs Merryman

JUNE 1991

Thesis Advisor:

Tung Bui

Approved for public release: Distribution is unlimited

92 3 10 134

92-06404



Unclassified

SECURITY CLASSIFICATION OF THIS PAGE

REPORT DOCUMENTATION PAGE				Form Approved OMB No 0704-0188	
1a. REPORT SECURITY CLASSIFICATION Unclassified			1b. RESTRICTIVE MARKINGS		
2a. SECURITY CLASSIFICATION AUTHORITY			3. DISTRIBUTION/AVAILABILITY OF REPORT Approved for public release: Distribution is unlimited		
2b. DECLASSIFICATION/DOWNGRADING SCHEDULE					
4. PERFORMING ORGANIZATION REPORT NUMBER(S)			5. MONITORING ORGANIZATION REPORT NUMBER(S)		
6a. NAME OF PERFORMING ORGANIZATION Naval Postgraduate School		6b. OFFICE SYMBOL (If applicable) AS	7a. NAME OF MONITORING ORGANIZATION Naval Postgraduate School		
6c. ADDRESS (City, State and ZIP Code)  Monterey, CA 93943-5000			7b. ADDRESS (City, State, and ZIP Code)  Monterey, CA 93943-5000		
8a. NAME OF FUNDING/SPONSORING ORGANIZATION		8b. OFFICE SYMBOL (If applicable)	9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER		
8c. ADDRESS (City, State, and ZIP Code)			10. SOURCE OF FUNDING NUMBER		
			PROGRAM ELEMENT NO.	PROJECT NO.	TASK NO.
					WORK UNIT ACCESSION NO.
11. TITLE (Include Security Classification)  ELECTRONIC FILING EVALUATION OF TAX PROGRAMS					
12. PERSONAL AUTHORS GRETCHEN OCHS MERRYMAN					
13a. TYPE OF REPORT Master's Thesis		13b. TIME COVERED FROM _____ TO _____	14. DATE OF REPORT (Year, Month, Day) JUNE 1991		15. PAGE COUNT 77
16. SUPPLEMENTARY NOTATION The views expressed are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government					
17. COSATI CODES			18. SUBJECT TERMS (Continue on reverse if necessary and identify by block numbers)		
FIELD	GROUP	SUB-GROUP	electronic filing of tax returns, ROMC model		
19. ABSTRACT (Continue on reverse if necessary and identify by block numbers) The goal of this thesis was to evaluate the electronic filing capabilities of three tax programs that are currently available. The systems discussed are CPAid Master Tax Program, Drake 89 Personal Income Tax and Orrtax PC Program. The evaluation was based upon the Representation, Operation, Memory Aids, and Control (ROMC) model by Sprague and Carlson, 1982. The evaluation was concerned with two questions: 1) What features are inherent in the electronic filing module of the three tax programs and 2) What features are not provided by the systems that should be included. Two techniques were employed to compare the findings of the evaluation conducted. The methods used were the figures of merit and the electre method. Both of these techniques revealed CPAid as the best tax program among the three systems evaluated.					
20. DISTRIBUTION/AVAILABILITY OF ABSTRACT XX UNCLASSIFIED/UNLIMITED    SAME AS RPT    DTIC USERS			21. ABSTRACT SECURITY CLASSIFICATION Unclassified		
22a. NAME OF RESPONSIBLE INDIVIDUAL Tung Bui			22b. TELEPHONE (Include Area Code) (408) 646-2630		22c. OFFICE SYMBOL AS/Bd

Approved for public release: Distribution is unlimited

Electronic Filing Evaluation of Tax Programs

by

Gretchen Ochs Merryman  
Lieutenant, United States Naval Reserve  
B.S., Pennsylvania State University, 1985

Submitted in partial fulfillment of the  
requirements for the degree of

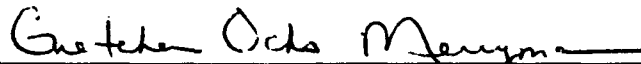
MASTER OF SCIENCE IN  
INFORMATION SYSTEMS

from the


NAVAL POSTGRADUATE SCHOOL


JUNE 1991


Author:

  
Gretchen Ochs Merryman

Approved by:

  
Tung Bui, Thesis Advisor

  
William Haga, Second Reader

  
David Whipple, Chairman  
Department of Administrative Science

## ABSTRACT

The goal of this thesis was to evaluate the electronic filing capabilities of three tax programs that are currently available. The systems discussed are CPAid Master Tax Program, Drake 89 Personal Income Tax and Orrtax PC Program. The evaluation was based upon the Representation, Operation, Memory Aids, and Control (ROMC) model by Sprague and Carlson, 1982.

The evaluation was concerned with two questions: 1) What features are inherent in the electronic filing module of the three tax programs and 2) What features are not provided by the systems that should be included.

Two techniques were employed to compare the findings of the evaluation conducted. The methods used were the figures of merit and the electre method. Both of these techniques revealed CPAid as the best tax program among the three systems evaluated.

<b>Accession For</b>	
NTIS GRA&I	<input checked="checked" type="checkbox"/>
DTIC TAB	<input type="checkbox"/>
Unannounced	<input type="checkbox"/>
Justification	
By	
Distribution/	
<b>Availability Codes</b>	
Dist	Avail and/or Special
A-1	

## TABLE OF CONTENTS

I.	INTRODUCTION . . . . .	1
II.	THE ROMC APPROACH . . . . .	4
A.	INTRODUCTION . . . . .	4
B.	REPRESENTATION . . . . .	5
1.	Human Interface . . . . .	5
a.	Question/Answer . . . . .	6
b.	Menu . . . . .	7
c.	Command Language . . . . .	7
d.	Adaptable Interface . . . . .	8
e.	Windows and Multiple Screens . . . . .	9
(1)	Screen Layout. . . . .	10
(2)	Triggered Changes . . . . .	11
(3)	Selection . . . . .	11
(4)	Copying . . . . .	12
(5)	Cloning . . . . .	12
(6)	User Perception of Windows . . . . .	13
f.	Icons . . . . .	14
(1)	Icon Advantages . . . . .	15
(2)	Icon Location . . . . .	16
g.	Graphics . . . . .	16
h.	Multimedia . . . . .	17
C.	CONTROL . . . . .	18
D.	OPERATIONS . . . . .	19
1.	Modeling . . . . .	20
a.	Network Models . . . . .	21
b.	Decision Theory Models . . . . .	21
c.	Human Information Models . . . . .	21
d.	Computer System Models . . . . .	22
e.	Other Model Representations . . . . .	22
f.	Model Base . . . . .	23
g.	Model Storage . . . . .	24

E.	MEMORY AIDS . . . . .	25
1.	Data Base . . . . .	25
a.	Data Base Models . . . . .	26
	(1) Record Model . . . . .	27
	(2) Relational Model . . . . .	27
	(3) Hierarchical Model . . . . .	28
	(4) Network Model . . . . .	28
b.	Data Base Model Selection . . . . .	29
2.	Views . . . . .	29
3.	Workspace, Libraries, and Links . . . . .	30
4.	Triggers and Profiles . . . . .	30
III.	TAX PACKAGE SPECIFICATIONS . . . . .	31
A.	INTRODUCTION . . . . .	31
B.	FUNCTIONALITY REQUIREMENTS . . . . .	31
C.	SYSTEM REQUIREMENTS . . . . .	32
1.	Interface . . . . .	32
2.	Control . . . . .	32
3.	Supplemental features . . . . .	33
D.	THE ROMC APPROACH AND ELECTRONIC FILING REQUIREMENTS . . . . .	33
1.	Representation . . . . .	33
2.	Operations . . . . .	34
3.	Memory Aids . . . . .	35
4.	Control . . . . .	35
E.	CONCLUSIONS . . . . .	36
IV.	ELECTRONIC FILING EVALUATIONS . . . . .	38
A.	THE ROMC APPROACH AND ELECTRONIC FILING SOFTWARE . . . . .	38
1.	Representation . . . . .	38
2.	Operation . . . . .	38
3.	Memory Aids . . . . .	39
4.	Control . . . . .	39
B.	CPAID TAX PACKAGE . . . . .	39
1.	Identified Representation Features . . . . .	40
2.	Desired Representation Features not Provided . . . . .	41
3.	Identified Operation Features . . . . .	42

a.	Intelligence . . . . .	42
b.	Design . . . . .	42
c.	Choice . . . . .	42
4.	Desired Operation Features not Provided . . . . .	42
5.	Identified Memory Aid Features . . . . .	43
6.	Identified Control Features . . . . .	44
7.	Desired Control Features not Provided .	44
8.	Conclusions . . . . .	45
C.	DRAKE TAX PACKAGE . . . . .	45
1.	Identified Representation Features . . .	46
2.	Desired Representation Features not Provided . . . . .	46
3.	Identified Operation Features . . . . .	47
a.	Intelligence . . . . .	47
b.	Design . . . . .	48
c.	Choice . . . . .	48
4.	Desired Operation Features not Provided	48
5.	Identified Memory Aid Features . . . . .	49
6.	Identified Control Features . . . . .	50
7.	Desired Control Features not Provided .	50
8.	Conclusions . . . . .	50
D.	ORRTAX SOFTWARE PACKAGE . . . . .	51
1.	Identified Representation Features . . .	51
2.	Desired Representation Features not Provided . . . . .	51
3.	Identified Operation Features . . . . .	53
a.	Intelligence . . . . .	53
b.	Design . . . . .	53
c.	Choice . . . . .	53
4.	Desired Operation Features not Provided	53
5.	Identified Memory Aid Features . . . . .	54
6.	Identified Control Features . . . . .	55
7.	Conclusions . . . . .	55
V.	EVALUATION TECHNIQUES . . . . .	56
A.	FIGURES OF MERIT . . . . .	56
B.	MULTICRITERIA DECISION AID . . . . .	58

1. The Electre Method . . . . .	58
2. Tax Software Evaluation . . . . .	61
VI. CONCLUSIONS . . . . .	65
LIST OF REFERENCES . . . . .	67
INITIAL DISTRIBUTION LIST . . . . .	68



## LIST OF TABLES

TABLE I.	TAX SOFTWARE TABLE OF MERIT CALCULATION . . .	59
TABLE II.	GRADING SYSTEM POINTS . . . . .	62
TABLE III.	TAX PROGRAM SCORES . . . . .	62
TABLE IV.	EVALUATION RESULTS . . . . .	64

## **I. INTRODUCTION**

The number of tax returns that are submitted for electronic filing is increasing each year. In 1986 there were five participating firms and 25,000 returns filed electronically. These numbers increased to 9,429 and 1,200,000 respectively in 1989. According to the Internal Revenue Service (IRS), this growth of electronic filing participation can be attributed to two factors: 1) Electronic returns take a much shorter time to process, and 2) Electronic returns have a much higher accuracy rate than manually prepared returns (IRS Publication 1345, 1989).

Due to the increased participation in electronic filing, the number of software tax packages available to perform this function has increased. The number of software packages available makes it difficult for tax preparers to choose a system that meets their needs.

The goal of this study is to evaluate the electronic filing capabilities of three tax programs that are currently available. The systems discussed are CPAid Master Tax Program, Drake 89 Personal Income Tax and Orrtax PC Program. The evaluation will be based upon the Representation, Operation, Memory Aids, and Control (ROMC) model by Sprague and Carlson, 1982.

Representation consists of the features used to assist a user in conceptualization of a system (e.g. a graph showing transmitted returns versus rejected returns). Operations are used to support the intelligence (the ability to search the environment for conditions that call for decisions), design and choice of a system. An example of an operation is validating a client's return for electronic filing. Memory aids are used to support representations and operations of a system. Internal and external databases for storage of gathered information are example of memory aids. The control aids are used to help a user control the representations, operations, and memory aids. Two examples of control aids are on-line help and error messages. (Sprague and Carlson, 1982, p. 96).

This evaluation will be concerned with two questions: 1) What features are inherent in the electronic filing module of the three tax programs and 2) What features are not provided by the systems that should be included. The DSS design specifications of a tax program's electronic filing module will be discussed.

The outline for presenting this thesis is as follows: Chapter II discusses the ROMC approach, its basic characteristics and the features associated with each area; Chapter III discusses the DSS design for tax package specifications of the electronic filing component; Chapter IV discusses the evaluation of the electronic filing module of

the three tax packages; Chapter V discusses the different evaluation techniques used to summarize the findings; and Chapter VI is the conclusion.

## **II. THE ROMC APPROACH**

### **A. INTRODUCTION**

The design approach that this thesis is based on is the Representation, Operation, Memory and Control aids (ROMC) approach (Sprague and Carlson, 1982, p. 15). This approach provides four areas (ROMC) used to identify specific user requirements. This identification is intended to reduce the gap between a systems capabilities and actual user requirements. For example, a user requirement may be to show the relationship between total returns processed and returns filed electronically. A corresponding system capability would be a graph of these two variables. By identifying requirements, a system can be designed to correctly support a user.

The variety of levels and types of decision making processes supports the use of the ROMC approach. This approach is a process independent model that allows a DSS to be analyzed and designed to support the cognitive styles of many users. The type of decision maker and the nature of a problem determines the type of decision process.

## **B. REPRESENTATION**

"Decision makers have trouble describing a decision-making process, but they do seem to rely on conceptualizations, such as pictures or charts, when making or explaining a decision."

(Sprague and Carlson, 1982, p. 98).

From a user's point of view, the representations help communicate or conceptualize a problem or situation. A decision maker may not be able to describe the process of decision making but he can conceptualize it through such things as tables, charts, and graphs. These representations must be in forms that are familiar to a user. A user should not be forced to deal with representations, such as tables or charts that do not correspond to his or her conceptualizations.

### **1. Human Interface**

The software user interface should provide a user with a friendly way to enter and receive information. The capabilities and flexibility of an interface determines the success of an application. Users cognitive styles and decision making processes must be incorporated in an interface. This incorporation is done through a representation of a design.

An interface should have the capability to provide graphics to support representations defined by users. Graphic displays combined with text frequently are much better information communicators than text alone (Grittins, 1986, p.

527). The graphics must be in the form desired by a user and have the capabilities to be modified by a user.

To examine the different levels and types of representation, the following types of interfaces will be discussed: question/answer, menu, and command language.

**a. Question/Answer**

The question/answer type of interface gives a user little control over operations performed. The computer system asks questions and a user provides answers. The dialogue is therefore one-sided with the system actively controlling the direction of the interactive session. The little control a user does have lies in the responses he gives. The answers provided implicitly direct the sequence of further questions. However, the sequence and questions asked are usually relatively fixed.

Because a user cannot control the interactive session, if a change is desired one must go through the entire iteration or abort the process and start over. Either option is frustrating to an individual. An abbreviated version versus the full text mode is one way to make the interaction more flexible. Because of the inflexibility and dominant system control, the question/answer interface is most often appropriate for an inexperienced user performing structured tasks. (Bennet, 1983, p. 67)

### **b. Menu**

The menu interface gives a user more control than the question/answer format. The menu provides a short list of alternatives from which a user can choose in communicating to a system. This list of alternatives can determine the nature of the operation and the data on which an operation is to be performed. The menus are grouped and hierarchically structured in a way that can be easily recognized by a user. There is however, still not much demand placed upon a user. The menu items should be clearly stated, understandable, and easily found in the list. All a user is required to do is select an item and be familiar with the hardware used for selection (keyboard, mouse, touch panel, etc.). The simplicity and finite capability of menus therefore make them appropriate for the inexperienced computer user who is performing structured tasks.

### **c. Command Language**

To provide a user control over the representation, operations, and memory aids, command language interface can be used. This interface consists of a set of noun verb statements that control the operation of the system. The commands issued during an interactive session are initiated by the user, thus enabling control over the system. Through these commands the user controls the system operations that are performed, the



data that is used for these operations and the way the data output is presented.

This interface provides a user with a greater latitude in the sequence of operations. More power and flexibility is provided with this interface. However, the command language must be learned, increasing the demand upon a user. There is a tradeoff between the power and flexibility of the interface and the amount of knowledge needed to use it. This interface should therefore be employed for users who have more computer experience.

#### **d. Adaptable Interface**

As a user becomes more experienced in operating computers, the type of interface that can successfully be implemented changes. Therefore, there is a demand for the interface to be adaptable to the various user computer skills. This can be done through a combination of the above interfaces. One way is to have a continuous interactive session with distinctive phases, each having its own dialogue and type of interface depending upon the computer skills of a user.

Another way to make the interface adaptable is to provide a mixture of interfaces within each phase. The following is an example of combining a question and answer session with menus. Computer provided question: "What returns would you like to transmit?" Menu provided containing a list

of answers for a user to choose from: All returns, marked returns, returns dated March 15 1991, returns prepared by Jones.

Because of the variety of styles and levels of users it is almost impossible to develop a system that would satisfy everyone. Therefore the best system must be designed to be flexible, user friendly and provide a variety of options for many users.

#### **e. Windows and Multiple Screens**

"Over the last 20 years the development of interactive computer systems has progressed to the stage where capabilities of the user's device is often greatly in advance of its exploitation by dialogue designs." (Grittins, 1986, p. 521)

User interface design remains a challenge for the developers of computer software. Exploration of innovative tools to restructure the way information is presented and controlled is a complex task. One of the tools that displays information in an innovative way is windows. This tool displays menus, files and/or other objects on the screen simultaneously.

The concept of windows is to have a computer screen partitioned into discrete sections, each containing different types of information. For example, a spread sheet may present the working tables in one window and a set of available commands in another. Windows can also be arranged to overlap one another in order to provide additional information to the

user simultaneously. Each window is linked in some way depending on the task. This interface increases the capability of presenting a user's representation and perception of information.

(1) *Screen Layout.* The layout of a screen using multiple windows is an attempt to represent the information in a visual form, which aids in increasing a user's understanding of the application. The idea is to increase the users visual scope,

"the degree to which the user is able to integrate information across a display of multiple windows and screens and to grasp the whole of whatever is being displayed." (Norman, 1986, p. 230)

In order for increased perception to occur, windows must be presented in a way that is logical and familiar to a user. If this is not accomplished, or if irrelevant and distracting information is presented, the visual scope may actually be decreased. This again enforces the importance of matching and displaying a user's desired visual and graphical representations with the systems capabilities.

"When there is a mismatch between the user's visual expectations about the pattern of display and the actual operation of the system, it is hypothesized that the performance will be impaired." (Norman, 1986, p. 231)

The information that appears on the screen is referred to as the surface layout and the user's visual conceptions are referred to as the mental layout.

Using windows enhances a user's representation in two ways. First by displaying more information simultaneously, through the use of a partitioned screen for multiple windows. And secondly through the use of overlapping windows. In this case more information is not simultaneously displayed but users infer that it is there. The use of windows increases the chances of making a match between the system's capability of presenting representations and a user's expectations (Norman, 1986, p. 231).

(2) *Triggered Changes.* Different ways to improve the similarity between a system and a user in the area of representation is through the use of triggered changes, selection, copying and cloning. Triggered changes allows a user to make changes to the contents of information on one screen which produces changes in information on a linked screen (Norman, 1986, p. 235). An example of this is to have two screens, one with client personal information and one with a client summary letter. By changing the address of the client on the personal information screen, the address automatically changes on the summary letter screen.

(3) *Selection.* Another way to increase the similarity of representation between a user and a system is through selection. The selection process uses one screen as the working screen. An item is then selected from this screen and detailed information about it is displayed on another

screen without changing the working screen. An example would be to have a list of clients on the working screen. A particular client could be selected and detailed information about the client (date return filed, date billed, date payment received) would appear on another screen. This not only provides a user with more information, but also provides instant feedback on a user's selection.

(4) *Copying*. Copying is another way to increase the possibility of making a match between the representation of a user and a system. Copying allows a copy of the working screen to be made and displayed on another screen. This screen could then be used for future reference, creating a constant visual reminder of its contents without having to produce a printed copy. An example of this is to have a copy of the list of clients made from the working screen. Other operations could be done in the working screen while still having the list of clients in the other screen.

(5) *Cloning*. One final example to illustrate the possibility of improving the similarity between a user and a system in the area of representation is through cloning. Cloning is similar to copying in the sense that the working screen is copied. However, in cloning the copied screen now becomes an additional working screen. This allows you to simultaneously perform an application on the screen with different inputs. The two outputs would then be displayed at

the same time, allowing a user to instantly compare the results of the different inputs. An example of this is to have a screen that shows the amount owed by the client or the amount to be refunded to the client. Different inputs could be used for each screen, such as amount of IRA contribution, and the results would appear simultaneously. This would be a useful tool for tax planning.

(6) *User Perception of Windows.* The use of windows and multiple screens increases the overall representation of a user. When more than one window is displayed at a time a user can perceive them in a number of different ways. First, one can perceive them as simultaneous screens, reading them from top to bottom and left to right. An example of this would be displaying continuous pages of a document on partitioned windows of a screen.

Another way to comprehend a screen layout is through integration. Each window provides different information and a user extracts various bits of this information to form a conclusion or provide a base for future applications. An example of this would be to have one screen showing the forms and schedules used in preparing a client's return. Another screen would show the amount owed by the client or the amount to be refunded to the client. A tax preparer would then use necessary information from each screen to determine if the client was eligible for electronic filing

(only certain forms and schedules and returns having money being refunded to the client are eligible for electronic filing).

Another way to perceive a screen layout is through selective attention. A user may only attend to one screen at time, but switch screens randomly when desired. A user filters the screens and attends to the one that is requiring the most attention depending on the level of computer processing. This allows for real time interaction between a user and a system. An example of this is to have one screen contain current information concerning the status of returns filed electronically. The other screen would be used to continue processing a client's return. A preparer would continually be updated concerning the status of electronically filed returns.

Windows and multiple screens enhance the representation of a user. One must however, not overlook the fact that a representation must be explicitly defined by a user in order to construct an appropriate match between user and system.

#### **f. Icons**

Icons in the user interface of software packages is common. Icon interface uses pictographic symbols in order to represent underlying objects in a computer system. These pictographic representations of data have been used predominantly to replace question/answer, menu and command

interfaces as the means by which computer user dialogue is supported.

An icon may represent processes or data and their attributes, associations, or states. Through this technique a user selects one or more icons to invoke specific processes and manipulate data. Some icons are used only as visual indicators of existing processes, while others dynamically change to reflect changes in a process. By using icons that correspond to actual objects, a user friendly dialogue can be created with little demand or training required from a user.

(1) *Icon Advantages.* As mentioned above, one of the main advantages of using an icon user interface is the ease of use. There is no prerequisite for a user to have detailed knowledge of data or processes.

"The use of metaphors and the extra dimension of graphical form can both be used to exploit the operator's ability to infer function and attribute from a pictographic symbol."  
(Grittins, 1986, p. 526)

Regular use of a system using an icon interface can improve performance of interactive tasks. The use of graphics also allows common attributes of objects to be represented by common graphical components.

Another advantage is in the form of representation. Graphic displays can convey information to a user much better than any other medium (Grittins, 1986, p. 527). Also, by using icons, the physical display area of a screen is much more efficiently employed.



The aim of icon interfaces is to create a display environment in which a user can identify an object from its icon. The object identified can either be data or processes. Through this identification an icon is mapped to an underlying computer application.

Another advantage of icons is the use of existing metaphors. Through the use of metaphors, a user can determine attributes of a system's objects by associating icons with familiar physical objects and their attributes. This expands the possibility of incorporating a user's representations into a system.

(2) *Icon Location.* The location and movements of icons in the display area is important. The most advantageous way to display icons is in cells. Through the use of cellular display, similar icons can be grouped together in batches. By doing this, logical functions are filtered into chunks making it easier for a user to find the desired icon (Grittins, 1986, p. 533).

#### ***g. Graphics***

Inherent in the use of windows and icons is the implementation of graphics. Most systems in the near future will use a graphical user interface. Use of graphics increases a user's representation and the overall ease of application use. The need for a user to have information displayed in a

way that is easily recognized is increasing the demand for applications to use graphic displays.

The advantages of implementing a graphical user interface include increased application, ease of use, quicker learning by a user, and greater functionality. As systems increase in complexity the biggest advantage is the reduction in effort required to generate new solutions to specific situations. The use of graphics enables a wider range of computer users to produce effective results with little training or difficulty.

The use of graphics increases the level of user representation. The complexity of a system can now be transparent to a user. This allows the inexperienced user to successfully employ computer applications and increases the extent to which these applications can be effectively implemented.

#### **h. Multimedia**

Combining interactive computers with full motion video and compact disk sound is the future computer interface (Lippincott, 1990, p. 215). The use of this multimedia interface will allow computers to reach even more users by further reducing the complexities of powerful applications.

Within the multimedia environment, the area that will be employed in greatest detail is the area of video.

Existing graphics combined with video will become a new interface, the video user interface.

"Windows will be filled with stills and motion video, high resolution icons will become animated graphics and audio will be a standard accompaniment to text." (Lippincott, 1990, p. 216)

By combining graphical, video and audio information, a user perceives the representation of information in a more realistic manner. The use of multimedia will improve training and increase the capabilities of the help function. By using voice as a medium, a level of interaction and feedback can be provided that could not be given with only a paper/on line manual.

Another area that will be greatly enhanced is networks. The performance of PC presentations, electronic mail, workgroup tasks, video conferencing and personal information managers will be greatly enhanced. Through the use of multimedia, the match between a user's representation and computer's ability to present these representations will increase.

### **C. CONTROL**

The type of interface employed will also define the amount of control a user has over a system. The control mechanisms enable a user to make use of the representations, operations, and memory aids. There are various types of control mechanisms that are supported by the different interfaces. The type of

control used depends upon the style, knowledge and skill of an individual user.

The control aids consist of mechanisms used to facilitate the use of a system. One way of implementing control is to provide menus or function keys to invoke various system operations which are enforced across various representations. Another way to facilitate the use of a system is to include support aids for training and explanations on system use. Examples include natural language error messages, on-line training that employ the user to learn by doing, and help commands. A third type of control mechanism is operations that allow a user to change or override system default values. For example, if a system provides a graph showing tax preparers weekly statistics with default scale and axes labels, it should allow a user to change the default values.

#### **D. OPERATIONS**

Operations is the ability to analyze and manipulate a user's representations. The operations are classified into three categories for decision making; intelligence, design, and choice.

Gathering data, identifying objectives, diagnosing the problem, validating data and structuring the problem are operations associated with the intelligence category. Gathering and manipulating data, quantifying objectives, generating reports and alternatives, and assigning risks or

values to alternatives are design operations. Choice operations include generating statistics on alternatives, simulating results of alternatives, explaining alternatives and choosing alternatives.

### **1. Modeling**

These operations are used in the modeling component of an application.

"It is the modeling component that gives decision makers the ability to analyze the problem fully by developing and comparing alternative solutions." (Sprague and Carlson, 1982, p. 257).

The modeling component therefore is the main tool for supporting the activities of users when making decisions and solving problems. Some of these activities include deduction, simulation, analysis, projection and suggestion and comparison of alternatives.

The intended purpose of a good model is to verify user requirements by discussing what a system should and should not do. There are several modeling methods that can be used when developing a system (Andriole, 1989, p. 57). Some of them include network models, control theory models, decision theory models, models of human information processing models, and computer systems models. When choosing one of these models attention must be given to the similarity between application requirements and model characteristics.

### **a. Network Models**

Network models consider a computer system and a user to be equal elements in the over-all process.

"The individual task performed by both the user and the system are described in terms of expected performance and in terms of logical predecessor-successor relationships." (Andriole, 1989, p. 60)

The network of tasks are defined as a performance model of a user-computer system. In control theory models a user is considered as an element in a feedback control loop. The basis for these models include statistical estimation, decision theory, and control theory.

### **b. Decision Theory Models**

A user is considered an important aspect in the decision theory model. The main concern of these models is the decision-making behavior of a user. Two things are required for this model; 1) a set of possible states and their estimated possibilities and 2) a set of possible decisions and their expected values and cost in the various possible states. Courses of action are selected by comparing various costs and values of each possible state. These models are successful for suggesting optimal decisions and describing the decision-making behavior of users.

### **c. Human Information Models**

The models of human information processing include intensive analysis of the problem to be solved. Thorough examination of the task environment, the problem space used to

represent the problem, and the procedure used to achieve a solution is required. This type of model is rarely used because most applications are too detailed and task oriented.

#### **d. Computer System Models**

In computer system models the computer is the main component of the design. The behavior of the computer component of an interactive system is described, but user performance details are not. These type of models are good for determining whether or not user requirements are being met in regards to system performance measures. They are however of little help in determining what user requirements are.

#### **e. Other Model Representations**

Other ways to represent applications include narration, flowcharting, and screen display methods. Narration can be a powerful communication tool. It however, is best used when the application is relatively uncomplicated. Narration should include what the system will do, indicate and describe the input and output requirements, and suggest software/hardware configuration. To be effective, narration should be combined with and illustrated by simulated screen displays.

Various types of flowcharting can be used to develop effective models of application processes. Conceptual flowcharts should be at the basis of a system representing the pictorial presentations of the flow of information. Functional

and logical flowcharts can be used to describe the flow of data through the system, and the location of decisions.

Simulation of screen displays are sometimes the most useful type of model. Screen displays allow a user to precisely see what a system can do. With computer aided screen displays a user can also inspect the interactive graphics capabilities of a system. This is very useful since it allows a user to see and inspect each part of the interactive sequence of an application.

#### **f. Model Base**

The modeling component of an application consists of several models which make up the model base. Within this model base will be various types of models. The purpose of these models are to support a variety of tasks and analysis approaches of a user.

The model base is an important resource within an application that must be managed carefully in regards to model creating, restructuring, editing and inquiry. In order for a model base to be utilized effectively, it must be linked with the dialogue and data component of an application. This allows interactive and real time processing. In order to provide output data the model must communicate with internal and external data sources, and transaction data.

Different models will be associated with the varying levels of decision making; strategic, tactical, and



operational. There will be various data flows for the models at each level. The capability for data to flow between each level is mandatory since each level must interface with the other levels.

#### **g. Model Storage**

There are many ways to store and represent each model. Three different types of storage include: 1) subroutines, 2) sets of modeling statements, and 3) models as data. The most common way for a model to be stored is as a subroutine. When using subroutines, the model is called as a subroutine of a main program or another module. Each subroutine accomplishes a specific task. Values or data needed are passed through parameters.

Another way to represent models is as a set of modeling statements. The modeling statements will have limited input and output functions and will be embedded in a sequence of problem solving activities which must be interpreted each time it is called.

The third way to represent models is to store the models as data. There are advantages to storing models in this representation. First, this representation makes it easier to see what the model does and what it did in the past. Second, it is easier to explain the operation of the model when it is stored as data. This helps to facilitate communication between

developers and users. And thirdly, when it is stored as data it is much easier to update the model.

Each model is viewed by a user as different operations that are used to manipulate representations. Therefore, it is important to include appropriate and required operations determined by users. If the models include irrelevant or useless operations the overall success and use of a system may decline.

## **E. MEMORY AIDS**

Memory aids are used to assist a user in linking the representations and the operations of a system. They provide short term memory aids and act as user help and learning tools. Several types of memory aids can be provided to support the use of the representations and operations. Some examples include data bases, views, workspaces, libraries, links, triggers, and profiles.

### **1. Data Base**

A data base is a collection of data from sources which a user thinks may be relevant to a decision. The data stored in a data base is maintained data not transient data and therefore is considered to be of relevant value to a user. The data base is an important aspect of an application. Without it very little processing could be accomplished unless tedious entry of information was entered each time.

Each application uses both internal and external data bases. Internally a system collects information that is used for future planning, control, and operation of the application. External data bases consist of information collected by an outside organization or application. Without these data bases, an application would have to include functions to collect and maintain the needed data. This would increase the complexity of an application and possibly decrease system performance.

Having a data base provides many advantages. These advantages include simplified collection and maintenance of data, simplified application design, elimination of conflicting performance and security requirements, and increase of possible data sharing. The data base not only acts as a storage place for application data, it also provides other memory aids including workspaces, libraries and links and triggers among data.

#### **a. Data Base Models**

A data base can be modeled a number of different ways. Each model provides a method of representing, organizing, storing and handling the data of an application. The components of a data base model include: 1) data structures (lists, tables, relations, hierarchies, and networks); 2) operations applied to the data structures (retrieval, update, combination and summation); and 3)

integrity rules defining the possible states for the data structures.

The record model, relational model, hierarchic model, and network model are different types of data base models. Each model describes a set of objects; values or relationships among the values. These objects are represented by fields and relationships are represented by a collection of fields called records.

(1) *Record Model.* The record model consists of fields combined into records, and the data base is a collection of these records. Operations associated with the record model include: 1) creating a record, 2) updating a field in an existing record, 3) deleting a record, and 4) selecting a record. Each record must have a field that is unique. This field is called the key field. Every field must contain a value and new record types cannot be added. (Sprague and Carlson, 1986, p. 227)

(2) *Relational Model.* In the relational model the data structures consist of relations which are fields that are related. Each relation is like a table having rows and columns. Each field has a domain that defines the allowed values for that field. The operations associated with the relational model include: 1) insertion of a row, 2) updating the value of a field, 3) deleting a row, 4) creating a relation, 5) deleting a relation, 6) selecting a row from a

relation, 7) joining two relations based on common values of fields, and 8) projection; selecting a subset of fields in a relation. The difference between the operations of the record model and the relational model are the constraints placed upon the operations of the relational model. (Kroenke and Dolan, 1988, p. 132)

(3) *Hierarchical Model*. The hierarchical model represents data relationships using hierarchies or trees. All data relationships must be transformed into hierarchies before they can be defined in the database. The structures in this model represent the information that is captured in fields of the relational model. Also, certain records must exist in the hierarchical model before others can exist. The operations associated with the hierarchical model include: 1) creation of a record, 2) deletion of a record, 3) updating a field in an existing record, 4) retrieving the next record within the same level, 5) retrieving the next record at the lower level and 6) retrieving the next record at the higher level. The operations of this model help navigate through the data structures. Constraints on the operations exist, but they are not as strict as the constraints on the operations of the relational model. (Kroenke and Dolan, 1988, p. 443)

(4) *Network Model*. The network model is similar to the hierarchical model. It consists of records and links among these records. The constraints of the network model are

weaker than those of the hierarchial model. (Brookes, Grouse, Jeffery and Lawrence, 1982, p. 196)

#### **b. Data Base Model Selection**

The type of data base model used to support an application depends on the operations and integrity constraints required. The specific users, decisions and capabilities of the Data base management system (DBMS) used should also be considered when choosing a particular data base model.

The DBMS provides operations used to create, maintain, and access the data base. The DBMS operations can be categorized into the following areas: 1) dictionary, 2) creation, 3) deletion, 4) update, 5) query, 6) views, 7) protection, 8) sharing, 9) recover, and 10) optimization.

#### **2. Views**

A view is another type of memory aid provided for the user. A view is a subset of data extracted from the data base which may be relevant to a particular process, application, or decision. A view customizes the data base for the application at hand. A view can join data from two or more tables within a data base, thus creating a new table needed for an ad hoc situation. It is a temporary subset of an existing database that is implemented through a query language. The use of views also serves the purpose of protecting sensitive data from unauthorized users.

### **3. Workspace, Libraries, and Links**

Workspaces are temporary memory aids which are used to store accumulating results of operations on representations. They are used as intermediate storage areas similar to a piece of scratch paper. If the information is needed for further processes it can be saved in a library. The library provides long-term memory for relevant information or results created in the workspace. Information in one workspace or library is often needed in another workspace or library. The ability should exist to transfer the information between them. This is done through the use of links. Links provide information needed to make associations between different workspaces and libraries.

### **4. Triggers and Profiles**

Triggers are memory aids used to automatically perform operations or to remind a user to invoke certain operations. They act as reminders to users that certain operations need to be performed. Profiles are used to store initial defaults and status data. They can be user specific based upon the desired representations defined.

### **III. TAX PACKAGE SPECIFICATIONS**

#### **A. INTRODUCTION**

A tax software package should successfully automate the process of preparing, processing, and filing tax returns. If an automated process provides the capabilities to complete these functions, a tax preparer's time is used more efficiently and the job is made easier. Certain specifications must however be inherent in a software package in order for a preparer to maximize the automated capabilities.

#### **B. FUNCTIONALITY REQUIREMENTS**

A software package must provide functions necessary to calculate an individual's Federal and State tax return. These functions include processing for: 1) Form 1040 2) All other IRS/State Forms and Schedules and 3) Depreciation and Depletion. Once an individual's return is calculated, the software must provide the capability for electronic transmission and filing. A system must also have the capability to print tax returns including individual forms and schedules.



## **C. SYSTEM REQUIREMENTS**

### **1. Interface**

The capabilities mentioned above are the functions required to complete an individual tax return. Having these functions does not ensure success of the system. A good interface must also be provided. An interface should provide a combination of different techniques in order to meet various styles of many possible users. As discussed in the previous chapter, a system should make use of basic features including: question/answer format (provides simplicity for a beginning user); menus which use text, graphics, and icons; and windows or multiple screens.

### **2. Control**

On-line help is a feature that should be provided to facilitate a tax preparer in using a system. This help should be as extensive and comprehensive as possible. A user should be able to access the help function from any point in a program and return to this same point when the help function is no longer desired. A simulation of the various features, using a simple tax return, should be provided (including data entry, calculation, electronic filing, printing, summary reports, and client letter). A tutorial should also be included as a basic feature.

### **3. Supplemental features**

A tax software package should also provide some supplemental functions that incorporate other tasks of a tax preparer into the program. Features that should therefore be provided include: Billing table (provides automatic fee calculation); Report Generator (provides summary information, statistics, grouping, and sorting of collected data); client letter (informing the client of tax refund or payment, preparer's fee and point of contact for questions).

### **D. THE ROMC APPROACH AND ELECTRONIC FILING REQUIREMENTS**

Being able to electronically file and communicate directly with the IRS or with a software company as a third party filer is another necessary feature. This enables a tax preparer or filer to utilize electronic transmission options. This also makes a software package marketable to a wider range of possible users.

#### **1. Representation**

Within the Electronic Filing module there are certain characteristics that should be included in the package. In the area of Representation, features that are necessary include:

1. Validation of client's return for electronic filing to prevent unauthorized or incorrect returns being submitted for transmission.
2. Explanations explaining why a return is ineligible for electronic filing. This assists a tax preparer in correcting a return and making it valid for electronic filing.

3. List of all clients eligible for electronic transmission. This provides a tax preparer with pertinent information in selecting clients for electronic transmission and acts as a double check reference for valid clients.
4. Validation of tax preparer as a qualified transmitter to eliminate the possibility of returns being sent as unauthorized transmissions.
5. List and status of returns transmitted to the IRS. This enables a tax preparer to track all submitted returns.
6. List of and errors noted for returns rejected by the IRS. This provides a tax preparer with the information needed to correct a return.
7. Explanation list of all IRS error codes. This provides a tax preparer with an on-line reference, thus making it much easier to correct invalid returns.
8. Graphical representation of actual transmission. This provides a tax preparer with a visual representation of the electronic transmission.
9. Summary and statistical reports. This provides a tax preparer with feedback and valid information for analysis and decision making.
10. Use of windows, multiple screens, graphics and icons. This increases the presentation of the representations, operations and control of a system.

## **2. Operations**

Features necessary in the area of Operations include:

1. Ability to gather needed tax data on clients and preparers for submission of return.
2. Validation of returns for electronic filing (based upon current IRS standards) to prevent invalid returns being transmitted.

3. Selecting and electronically transmitting returns (ability to operate using 1200, 2400, or 4800 baud modems) to successfully meet system requirements.
4. Transmitting test files to provide a tax preparer an opportunity to ensure the hardware is setup correctly.
5. Creating client transmission files in order for a return to be submitted successfully.
6. Creating acknowledgement and Declaration Control reports which are required by the IRS.
7. Preparing summary reports and statistics to provide a tax preparer with feedback and valid information for analysis and decision making.

### **3. Memory Aids**

Memory Aid functions that should be included in a software package include:

1. Internal and external Data Bases for storage of gathered information.
2. Desired views of these Data Bases to provide a tax preparer with needed information only.
3. Workspaces and libraries for each representation provided by a system for temporary and permanent storage of information.
4. Triggers to remind a user of required data or functions to assist a user in operating a system.
5. Profiles of a system including storage of any "reserved" words or default values to assist a tax preparer in successfully using a system.

### **4. Control**

Required functions necessary in the area of Control include:

1. Function keys to perform specific processes. This provides consistency and makes a system easier to use.
2. Natural language screen help and error messages which aid a tax preparer when there are problems or questions concerning the use of a system.
3. On-line help for all electronic filing functions to provide a tax preparer with a quick and easy reference for any problems encountered.
4. Simulation of electronic filing to provide a tax preparer with a visual representation of the required functions.
5. On-line tutorial for the electronic filing module which gives a tax preparer the capability of on hands training and provides a reference that can easily be accessed.
6. Capability to change system default values to customize a package to specific user needs.
7. Consistency in applying the above features which makes a system easier to use.

The control features may be the most critical in determining a system's ease of use. If these are designed poorly or not incorporated at all, a tax preparer may find a system difficult to use. With little or inadequate on-line help, a system will be harder to implement.

## **E. CONCLUSIONS**

An automated tax package can greatly enhance the productivity of a tax preparer. However, to do this certain features must be provided by a package. If some of the required features are not provided, a system can still be used, but maximum productivity will not occur. A systems's

success and overall use will be determined by features that are provided. Therefore when designing a tax software package, one must take into consideration the previously mentioned required features.

#### **IV. ELECTRONIC FILING EVALUATIONS**

##### **A. THE ROMC APPROACH AND ELECTRONIC FILING SOFTWARE**

The process used to evaluate the electronic filing component of the three tax programs discussed in this study consists of examination in these areas: 1) Representation, 2) Control, 3) Operations, and 4) Memory aids as discussed in chapters two and three. A brief description of each package will be provided followed by the identification of features available in each area. Desired features that are not available will also be noted.

##### **1. Representation**

The representation is the way a tax preparer conceptualizes an application of electronically transmitting client's tax returns and the operations associated with this. These representations provide the basis for evaluation and interpretation of the outputs derived from the operations of a system. They are often used as a very powerful communication tool for both a tax preparer and client.

##### **2. Operation**

Operation is the process of analyzing and manipulating the representations defined by a tax preparer. Operations can be grouped into three categories: 1) Intelligence, 2) Design and 3) Choice. An operation may be used in more than one

activity and may involve complicated decision aids and/or computation.

### **3. Memory Aids**

Memory aids are used to support the representations and operations of a tax preparer. They assist in linking the two areas together. Memory aids can be represented in a number of different forms including data bases, views, workspaces, libraries, links, triggers and profiles.

### **4. Control**

The control mechanisms are used to help a tax preparer make use of a system's representations, operations, and memory aids. This area may be the critical area for acceptance and use of a system. The control mechanisms are provided to handle and use an entire system.

## **B. CPAID TAX PACKAGE**

CPAid Tax Preparation Software is a comprehensive package that provides a preparer with an abundant capability to automate the tax preparation process. The following modules are included as part of this package: 1040 processing; Depreciation/Depletion; K-1/Partnership; Printing Tax Forms; Supplemental Programs; State Program; 1040X Amended; Electronic Filing; and Utilities.



## **1. Identified Representation Features**

The representation features identified were:

1. A list of clients that prequalified for electronic filing transmission.
2. A list of preparers designated to transmit returns electronically.
3. A list of Declaration Control Number's and their associated Taxpayers.
4. A list of all IRS error codes.
5. A cumulative list of all Declaration Control Number's for clients whose returns have been accepted by the IRS.
6. A list of reasons explaining why a return is not eligible for electronic filing.
7. A list and status of returns that have been electronically sent and acknowledged by the IRS.
8. Simulation of the actual electronic transmission.
9. Status of the electronic transmission.
10. A summary report of returns acknowledged by the IRS.

In order to provide these different representations a tax preparer is prompted to select various functions through the use of a menu interface. In performing certain functions a combination of menu and question/answer interface is used. Information is entered through the use of tabular forms similar to worksheets that are used during manual preparation.

## **2. Desired Representation Features not Provided**

Use of a menu interface limits the flexibility of a tax preparer. The system does not provide the use of windows or multiple screens. A tax preparer could increase performance capabilities if this technique were used. For example, if windows were incorporated into the system, the selection technique could be used. An example of this would be to have the summary report of returns acknowledged by the IRS as the working screen. Selection of desired returns would then be made. The detailed information of each selected return would appear on another screen without changing the summary report.

The use of a graphical user interface is another feature that could be provided by the system. With a graphical user interface a tax preparer could be provided with functions that increase the match between what he/she conceptualizes and what the system provides. For example, if a graphical user interface was incorporated, a tax preparer could be provided with various graphs representing numerical data such as: number of returns transmitted daily; number of returns accepted; number of returns rejected; percentage of tax returns electronically transmitted, etc. The use of graphics would increase the communication and information provided by the system for a tax preparers' evaluation and interpretation.

To increase the system ease of use, icons could be incorporated. Pictographic symbols could be used to represent various functions; printing reports, creating files, and

updating client/preparer lists. This would make it easier for tax preparers who are just learning to operate the system.

### **3. Identified Operation Features**

The operation features identified were:

#### **a. Intelligence**

1. Gathering tax data on clients
2. Gathering data on tax preparers
3. Validating returns for electronic filing
4. Selecting clients desired for electronic transmission
5. Transmitting returns electronically

#### **b. Design**

1. Creating client transmission files
2. Creating Declaration Control Number reports
3. Creating acknowledgement reports
4. Creating summary acknowledgement reports

#### **c. Choice**

1. Simulation of actual transmission
2. Explanation of errors in electronic return transmissions

### **4. Desired Operation Features not Provided**

The desired features that are not provided by the system are in the area of choice operations. The system should

be able to calculate summary statistics such as: percentage of returns filed electronically by preparer, region, office, etc; percent of returns electronically filed correctly; percentage of returns electronically filed having a direct deposit; etc. These summary statistics could then be used in the graphic representations previously mentioned.

## **5. Identified Memory Aid Features**

The memory aid features identified were:

1. Internal data base for the entire software package; default values, IRS requirements and standards
2. External data base for the entire software package; entered by a tax preparer
3. Views associated with the electronic filing module; subset of the system data base
4. Workspace for each representation previously provided
5. Library for each representation previously provided
6. Triggers:
  - a. Reminder that acknowledgement must be received before new transaction can be sent
  - b. Reminder that validation/revalidation must be completed before selecting a client
  - c. Reminder to recover after interrupted process
7. Profiles:
  - a. Storage of reserved words to perform operations
  - b. Storage of default values

The memory aids provided by the software package are comprehensive and enable a tax preparer to adequately perform all necessary functions.

#### **6. Identified Control Features**

The control features identified were:

1. Use of menus to display operations
2. Functions keys to perform certain processes required for electronic filing
3. Consistent means of selecting menu options (arrow keys, carriage return, beginning letters)
4. Use of natural language screen help and error messages
5. Additional on-line help function for certain areas
6. Ability to change default values

The control features of the system are consistent and helpful. The same function keys are used throughout the entire module to perform various operations. For selection of menu items, three ways are provided, each with the same result. This helps to satisfy the different styles of various tax preparers.

#### **7. Desired Control Features not Provided**

One missing feature in the control area of the system is the ability to have on line help provided for all operations. It is provided in certain areas but not in others. This inconsistency may cause a tax preparer to inadequately or

infrequently use the help functions that are provided. This may result in a negative attitude towards use of the system.

## **8. Conclusions**

The CPAid system provides a tax preparer with adequate features to perform the task of electronic filing for tax returns. The features provided have been in existence for a number of years. It lacks, however, in the use of newer technology such as graphics, multiple screens and windows. Incorporating newer technology into the system may increase the ease of use of the application. This may also enable a broader range of individuals to successfully apply the package.

## **C. DRAKE TAX PACKAGE**

Drake Software tax package is an automated program that allows a tax preparer to prepare, calculate, and transmit tax returns using a computer. This system utilizes two different entry modes; a full screen entry and a heads down entry. The full screen entry allows a user to enter data directly into the fields shown on the screen. When using the heads down entry a tax preparer enters data into a box in the lower left hand corner of the screen using key numbers found on the interview sheets. This type of entry eliminates a tax preparer from entering unnecessary fields for a specific return.

The Drake tax package provides the following functions:  
Set up (preparer information, form defaults, electronic filing

information, macros, result and update letters); 1040 entry and calculation; Electronic Filing; and Utilities.

### **1. Identified Representation Features**

The representation features identified were:

1. Status of each return electronically filed
2. A list of reasons explaining why a return is not eligible for electronic filing
3. Simulation of the actual electronic transmission.

The different representation features are presented to a tax preparer through the use of menus. However, after each selection is made and completed in the electronic filing submenu, the program returns to the main menu. A tax preparer must again select the electronic filing submenu to continue working in this area. This is a redundant feature that becomes tedious and time consuming.

### **2. Desired Representation Features not Provided**

The menu interface limits the flexibility of the system. The package does provide multiple menus on the screen simultaneously, but as previously mentioned, a user always returns to the main menu after completing a process. In addition to this inflexibility, there are numerous representation features that are not included. The desired representation features that should be provided by the package are:

1. A list of clients that qualify for electronic filing
2. A list of preparers designated to transmit returns electronically
3. A list of Declaration Control Number's for clients whose returns have been accepted by the IRS
4. A list of all IRS error codes
5. A list of returns electronically filed and accepted by the IRS
6. A list of returns acknowledged by the IRS

These features are necessary for a tax preparer to adequately track and manage his work. The system provides the operations needed to perform the actual electronic filing transmission, but does not adequately present the desired representations to a tax preparer.

The representations that are inherent in the system would be greatly enhanced if graphics and icons were used. This would make the system much easier to use, thus increasing the productivity of a tax preparer.

### **3. Identified Operation Features**

The following operation features were identified:

#### **a. Intelligence**

1. Gathering tax data on clients
2. Validating returns for electronic filing
3. Selecting clients desired for electronic transmission
4. Transmitting returns electronically



### **b. Design**

1. Creating client transmission files
2. Creating acknowledgement files
3. Creating status inquiries

### **c. Choice**

1. Simulation of actual transmission
2. Testing electronic transmission capabilities

## **4. Desired Operation Features not Provided**

The system lacks features in the areas of Intelligence and Choice operations. It does not provide any capability for gathering data on tax preparers. This operation would provide a security check for the system. Information would be gathered on tax preparers who are qualified to transmit returns electronically. If a tax preparer who is not qualified tried to transmit a return, the system would not complete the task. This would eliminate the possibility of returns being transmitting maliciously.

The system should also be able to calculate summary reports and statistics. An internal data base of information exists and is updated after every transmission, but no summary reports are provided. The capability of performing ad hoc inquiries on things other than individual client status is not possible. The information is in the system, but there is no

way to provide it to a tax preparer. It would be extremely beneficial to a tax preparer if the following information could be provided: 1) list of clients whose returns have been electronically filed; 2) status of electronically filed returns; 3) list of Declaration Control Number's and 4) summary statistics for electronically filed returns.

#### **5. Identified Memory Aid Features**

The memory aid features identified were:

1. Internal Data Base for the entire software package
2. External Data Base for the entire software package
3. Views associated with the electronic filing module; subset of the system Data Base
4. Workspace for each representation previously provided
5. Library for each representation previously provided
6. Triggers:
  - a. Reminder that printer must be on for Document Control Record to be printed
  - b. Reminder of any IRS files existing that have not been transmitted
  - c. Reminder of errors during transmission (out of memory, phone line busy)

The memory aids provided by the software package are adequate allowing a tax preparer to perform all necessary functions. As previously mentioned, information is adequately stored, it is just difficult to retrieve.

## **6. Identified Control Features**

The control features identified were:

1. Use of menus to display operations
2. Function keys to perform select electronic filing options

## **7. Desired Control Features not Provided**

The system lacks in the area of control. There is no on line help provided for any electronic transmission function. This reduces the capabilities and performance of tax preparers using the system, especially first time users. A tax preparer must struggle on his own to correct mistakes. The manual documentation provided is of some help. Information included in the manual should be incorporated into the system. This would make access to the information much easier. A list of IRS error codes should also be available to a tax preparer by means of a help function. This would reduce time required to get assistance when operating the system. The lack of help provided by the system may deter many inexperienced users from implementing and successfully using this software.

## **8. Conclusions**

The Drake system provides a tax preparer with the necessary features needed to perform electronic transmission of client returns. However, the lack of desired representation features may deter a tax preparer from continuing to use the

system. The inability for a tax preparer to get on-line help may also be a deterring feature of the system.

#### **D. ORRTAX SOFTWARE PACKAGE**

The Orrtax software package is a comprehensive automated program that enables a tax preparer to prepare, calculate and transmit client tax returns. It provides the following capabilities: 1040 processing, Electronic filing, Transmission only, State program, and Reports and inquires.

##### **1. Identified Representation Features**

The following were representation features identified:

1. A list of clients that qualify for electronic transmission
2. A list of preparers designated to transmit returns electronically
3. Declaration Control Reports
4. A list of and status of individual client returns
5. Status of electronic transmission

The above representations are chosen through the use of menus. Within certain areas, a combination of menu and question/answer interface is used. Only one menu or window appears on the screen at a time.

##### **2. Desired Representation Features not Provided**

The system provides most of the features desired by a tax preparer. It provides a list of all returns eligible for electronic transmission, however it does not provide any type

of error checking for individual returns. The only way for a tax preparer to manually check an individual return and find out why it does not qualify is to check the entire return for any mistakes or missing data. This is a rather tedious and time consuming task. It also requires a tax preparer to know every field required for electronic transmission.

This increased work by a tax preparer could be eliminated if error checking for individual returns was provided by the system. The system would check the return for errors. These errors would be reported to a tax preparer and the necessary changes could be made to enable a return to be filed electronically. This would reduce the time needed to check a return, thus increasing the number of tax returns that could be processed and filed electronically.

Along with this feature should be a list of all IRS error codes and an explanation of each. Even if the return is validated by the software, errors may still occur (eg. duplicate social security number). If a list and explanation of these errors were provided on line to a tax preparer the task of correcting the returns would be made easier. As with the other systems, the use of graphics and icons would increase the ease of use and quality of the software package.

### **3. Identified Operation Features**

The operation features identified were:

#### **a. Intelligence**

1. Gathering tax data on clients
2. Gathering data on tax preparers
3. Selecting, adding, or deleting clients desired for electronic transmission.
4. Transmitting returns electronically

#### **b. Design**

1. Creating client transmission files
2. Creating Declaration Control Reports
3. Creating acknowledgement reports
4. Creating client reports

#### **c. Choice**

1. Simulation of actual transmission
2. Electronic mailbox capabilities

### **4. Desired Operation Features not Provided**

The desired features that are not provided by the system are in the area of Intelligence and Choice operations. In the area of Intelligence, the system should be able to individually validate and check a return for electronic filing. This was previously mentioned under representation.

The lack of this feature reduces the capability of a tax preparer.

Features that should be provided in the choice operations include the capability of providing explanations of IRS error codes and calculation of summary statistics. The use of IRS error code explanations was mentioned under representation. By providing this feature, a tax preparer would reduce the time required to make corrections. The summary statistics would provide a tax preparer with useful information about his work.

#### **5. Identified Memory Aid Features**

The memory aid features identified were:

1. Internal data base for the entire software package
2. External data base for the entire software package
3. Views associated with the electronic filing module; subset of the system data base
4. Workspace for each representation previously provided
5. Library for each representation previously provided

The system provides a tax preparer with adequate memory aid features. The addition of triggers to remind a tax preparer of specific functions that needed to be completed should be included to eliminate possible mistakes.

## **6. Identified Control Features**

The control features identified were:

1. Use of menus to display operations
2. Use of function keys to perform certain processes required for electronic filing
3. Consistent means of selecting menu options
4. Use of natural language screen help and error messages
5. Additional on-line help

The control features provided by the system are consistent and helpful. One feature that could increase the control of the preparer is a list and explanation of IRS error codes. This would reduce the amount of time needed to make corrections and in turn increase the control of a tax preparer.

## **7. Conclusions**

The Orrtax Software package provides a tax preparer with the necessary functions needed to successfully transmit a return to the IRS. It provides useful reports and inquires for a tax preparer. The system is however, lacking two major functions: 1) the ability to validate individual returns and provide errors found and 2) the ability to provide a list and explanation of the IRS errors.



## **V. EVALUATION TECHNIQUES**

### **A. FIGURES OF MERIT**

Combining several evaluation criteria into a single decision variable is a technique that is often used when determining the effectiveness of a system. This technique is called figures of merit (Boehm, 1978, p. 223).

The figures of merit technique used here is weighted sum technique. To use this technique, the criterion by which to evaluate the different systems must first be determined. The criterion used to evaluate the three tax programs are representation, operations, memory aids, control and supplemental features.

Characteristics that comprise the representation criteria consist of:

1. list of clients eligible for electronic transmission;
2. list and status of files electronically transmitted to the IRS;
3. list of errors noted for returns rejected by the IRS;
4. explanation list of all IRS codes;
5. graphical representation of actual transmission;
6. summary and statistical reports;
7. use of windows, multiple screens, graphics and icons.

Characteristics that comprise the operations criteria consist of:

1. ability to gather tax data on clients;
2. validation of returns for electronic filing;
3. selecting and electronically transmitting returns;
4. creating acknowledgement and Declaration Control reports;
5. preparing summary reports and statistics.

The characteristics that comprise the memory aid criteria are:

1. internal and external data bases;
2. appropriate views of these data bases;
3. appropriate workspaces and libraries, triggers, and profiles.

The characteristics that comprise the control criteria are:

1. natural language screen help and error messages;
2. on-line help, tutorial, and simulation of electronic filing.

Once a criterion has been determined an evaluator assigns a weight to each. This weight reflects the relative importance of each criterion. For each system being evaluated, a rating is then given for every criterion. This rating reflects how well each system satisfies the associated criterion. Based on

the ROMC evaluation we assigned the following characteristics and ratings:

Excellent = 10;  
Good = 8;  
Average = 6;  
Fair = 4;  
Poor = 2;  
None = 0.

A weighted rating is then calculated for each criteria by multiplying the weight by the rating. These weighted ratings are then summed to provide a total score for each system. The results of this technique applied to the three tax programs (using the defined criteria) are shown in Table I.

The results lead us to rank CPAid tax software first, Orrtax second and Drake third. This, however, is a conclusion that is subjective and sensitive to the weights and ratings that we assigned. It however, does provide us with a way to pick out which weights and ratings are the most important. It is used here to show one possible evaluation technique for systems having several criteria.

## **B. MULTICRITERIA DECISION AID**

### **1. The Electre Method**

The multicriteria decision aid technique electre is another way to evaluate systems having multiple criteria (Bui, 1982, p. 157). This automated software procedure also takes subjective evaluation into consideration. Two coefficients are used in this technique; the condition of concordance and the

**TABLE I. TAX SOFTWARE TABLE OF MERIT CALCULATION**

CRITERION	WEIGHT	CPMD			DRAKE			ORRTAX		
		CHARACTERISTIC	RATING	WEIGHTED RATING	CHARACTERISTIC	RATING	WEIGHTED RATING	CHARACTERISTIC	RATING	WEIGHTED RATING
1. Representation	35	Good	8	280	Fair	4	140	Average	6	210
2. Operations	20	Good	8	160	Average	6	120	Fair	4	80
3. Memory Aids	15	Excellent	10	150	Excellent	10	150	Excellent	10	150
4. Control	25	Good	8	200	Fair	4	100	Average	6	150
5. Supplemental Features	5	Poor	2	10	Good	8	40	Poor	2	10
..... Total	100			800			550			600

condition of discordance. The condition of concordance requires the majority of criteria favor decision A.

The condition of discordance requires that no criterion favor decision B excessively. These two coefficients are used to make paired comparisons between different possibilities. The concordance coefficient indicates to what extent one option is better than another one. A perfect value for this coefficient is 1. The formula for this coefficient is:

$$C_{a/b} = \frac{\text{Sum on the weights of the criteria by which A outranks B}}{\text{Sum of the weights of all the criteria of the model}}$$

The discordance coefficient indicates the extent to which an option contains elements that might make it unsatisfactory. A "fatal" value for this coefficient is 1. The formula for this coefficient is:

$$D_{a/b} = \frac{\text{The greatest negative variation between the evaluation scores for a single criterion}}{\text{The maximum range between the highest possible score and the lowest possible scores}}$$

These coefficients are used in conjunction with thresholds chosen by a evaluator. The concordance threshold, P, ranges from 0.5 to 1, tightening the constraints as it approaches 1. The discordance threshold, Q, narrows the constraints as it approaches 0.

In addition to the concordance and discordance matrices, an outranking matrix is provided which shows the relationship among the different alternatives. The outranking matrix shows which alternatives outrank other alternatives, if

any. This provides a basis for selecting one alternative over another.

## **2. Tax Software Evaluation**

To apply this technique, the criteria for comparison must first be described. The criteria used to evaluate the three tax programs are as follows: 1) Representation, 2) Operations, 3) Memory Aids, 4) Control and 5) Supplemental Features. A description of these criteria was given in the figures of merit discussion. Weights must then be assigned by an evaluator representing the level of importance of each criterion. Based on the ROMC evaluation conducted we assigned the following weights:

Representation = 30;  
Operation = 27;  
Memory Aids = 10;  
Control = 30;  
Supplemental Features = 3.

A grading system must then be established for assigning grades to each system for each criterion. The points we assigned for the grading system are shown in Table II.

**TABLE II. GRADING SYSTEM POINTS**

	CRITERIA				
	1	2	3	4	5
Excellent	30	27	10	30	3
Good	23	20	8	23	3
Average	15	13	5	15	2
Fair	8	7	3	8	1
Poor	0	0	0	0	0

Table III shows the scores of the three tax programs.

**TABLE III. TAX PROGRAM SCORES**

	CRITERIA				
	1	2	3	4	5
CPAid	25	25	10	25	1
Drake	15	23	10	15	3
Orrtax	20	15	10	20	1

The following thresholds were used for the evaluation:  $P = 70$  and  $Q = 30$ . The results of this evaluation can be seen in Table IV. The concordance matrix, Table IVa, indicates the extent to which an option is better than another in terms of criteria weights. CPAid is better than either Drake or Orrtax in terms of concordance measures (97 and 100 respectively). The discordance matrix, Table IVb, indicates the extent to which a tax system contains a bad element that makes it unsatisfactory. Orrtax is the system having the greatest

discordance measures, both which are above the assigned threshold.

The outranking matrix, Table IVc, shows the system that satisfies both the concordance and discordance requirements.

The matrix reveals CPAid as being better than Drake or Orrtax. There is no outranking relation between Drake and Orrtax. The results stay the same when the thresholds are changed to  $P = 90$  and  $Q = 10$ . This indicates that CPAid is the best choice among the three systems.



**TABLE IV. EVALUATION RESULTS**  
**TABLE IVa. CONCORDANCE MATRIX**

	<b>CPAaid</b>	<b>Drake</b>	<b>Orrtax</b>	<b>#CI</b>
CPAaid	-	97	100	2
Drake	13	-	40	0
Orrtax	13	70	-	0

CI indicates the number of indexes satisfying P for each option

**TABLE IVb. DISCORDANCE MATRIX**

	<b>CPAid</b>	<b>Drake</b>	<b>Orrtax</b>	<b>#DI</b>
CPAaid	-	7	0	2
Drake	33	-	17	1
Orrtax	33	27	-	1

DI indicates the number of indexes satisfying Q for each option

**TABLE IVc. OUTRANKING MATRIX**

	<b>CPAid</b>	<b>Drake</b>	<b>Orrtax</b>
CPAaid	-	*	*
Drake	-	-	-
Orrtax	-	-	-

- \* indicates both concordance and discordance requirements satisfied
- indicates no outranking relations

## VI. CONCLUSIONS

The objective of this thesis was to propose a framework to evaluate the electronic filing capabilities of three tax programs. The evaluation was based upon the ROMC approach. A discussion of the literature revealed the different characteristics that comprise each of the components of the ROMC model: representation, operation, memory aids, and control. Based upon this information, the tax package specifications were determined for the electronic filing module of a tax program.

Three tax programs, CPAid, Drake and Orrtax, were evaluated. The evaluation revealed that:

1. All three of the systems were capable of performing the necessary operations needed to successfully complete all electronic filing requirements.
2. The area of control, specifically on-line help, was a weak point in the Drake system.
3. Individual validation of returns was not provided by the Orrtax system.
4. Little statistical information was provided by any of the systems.
5. None of the systems use windows, multiple screens, icons or graphics.

Two techniques were employed to compare the findings of the evaluation conducted. The methods used were the figures of

merit and the electre method. Both of these techniques revealed CPAid as the best tax program among the three systems evaluated.

The systems are all capable of performing the necessary functions of the electronic filing task. All of the systems, however, are lacking in the area of representation. The use of graphics and windows would enhance the ease of use and perhaps increase the success and use of the systems. Employment of these features is presently a common feature in most computer applications. The tax programs evaluated seem to be behind in employing the technological capabilities that are currently available. The importance of the user interface is revealed by Norman.

"The particular layout adopted by a user will drastically affect the user's understanding and expectation of events at the human-computer interface and could either greatly facilitate or frustrate the interaction" (1986).

Greater time is being spent on making computer applications user friendly. This is an area where designers of the tax programs evaluated must invest substantial time and effort. If this is neglected, other systems with better user interfaces will be used instead of the systems discussed.

## LIST OF REFERENCES

- Andriole, S. J., *Handbook of Decision Support Systems*, Tab Books Inc., 1989.
- Boehm, B. *Software Engineering*, Prentice-Hall, Inc., 1978.
- Brooks, C. H. P., Grouse, P. J., Jeffery, D. R., and Lawrence M. J., *Information Systems Design*, Prentice-Hall, 1982.
- Bui, T., *Executive Planning with Basic*, Sybex, 1982.
- CPAid Software, CPAid Inc., 1989.
- Drake Software, MCS/Drake Tax System, 1989.
- Grittins, D. "Icon-based Human-Computer Interaction," *International Journal Man-Machine Studies*, v. 24, 1986.
- Internal Revenue Service, *Revenue Procedure for Electronic Filing of Individual Income Tax Returns (Tax Year 1989)*, Department of the Treasury, 1989.
- Kroenke, D. M., and Dolan, K. A., *Database Processing Fundamentals, Design, Implementation*, Macmillan Publishing Company, 1988.
- Lippincott, R. "Beyond Hype," *Byte*, February 1990.
- Norman, K. L., Weldon, L. J., and Shneiderman, B., "Cognitive Layouts of Windows and Multiple Screens for User Interfaces," *International Journal Man-Machine Studies*, v. 25, 1986.
- Orrtax PC Program, Orrtax Inc., 1989.
- Sprague R. H. Jr., and Carlson, E. D., *Building Effective Decision Support Systems*, Prentice-Hall, Inc., 1982.

### INITIAL DISTRIBUTION LIST

- |    |  |   |
|----|--|---|
| 1. | Defense Technical Information Center<br>Cameron Station<br>Alexandria, Virginia 22304-6145                                     | 2 |
| 2. | Library, Code 52<br>Naval Postgraduate School<br>Monterey, California 93943-5002   | 2 |
| 3. | Tung Bui, Code AS/Bd<br>Administrative Science Department<br>Naval Postgraduate School<br>Monterey, California 93943-5002      | 1 |
| 4. | William Haga, Code AS/Hg<br>Administrative Science Department<br>Naval Postgraduate School<br>Monterey, California 93943-5002  | 1 |
| 5. | David Whipple, Code AS/Wh<br>Administrative Science Department<br>Naval Postgraduate School<br>Monterey, California 93943-5002 | 1 |
| 6. | Gretchen Merryman<br>7607 Bridgette Lane<br>Norfolk, Virginia 23518  | 1 |